<u>Attorney Docket No.: FUJI 17.634A (100794-00500)</u>

<u>IN THE UNITED STATES PATENT AND TRADEMARK OFFICE</u>
BOARD OF PATENT APPEALS AND INTERFERENCES

Appellant(s):   Ken'ichi IMAMATSU

Confirmation No.:   4605

Serial No.:   10/705,437

Filed: November 10, 2003

Title: **METHOD AND APPARATUS FOR UPDATING SOFTWARE IN RADIO TERMINAL DEVICE**

Examiner: William H. Wood

Group Art Unit: 2193

March 4, 2009

<u>BRIEF FOR APPELLANTS</u>

Board of Patent Appeals and Interferences
Assistant Commissioner for Patents
Washington, D.C., 20231

Sir:

      A Notice of Appeal was filed on September 4, 2008. Appellant hereby petitions for a

four-month extension of time, a petition pursuant to 37 C.F.R. 1.136(a) and authorization to

charge the requisite fee being enclosed. Appellant appeals to the Board of Patent Appeals

and Interferences from the Examiner's Decision, in the Official Action dated April 4, 2008,

finally rejecting claims 13, 17, 22, 26-27, 33-41, and 45. All requisite fees, including those

84344846_1

for this Brief set forth in 37 C.F.R. §41.20(b)(2), may be charged to Deposit Account No. 50-1290.

(i)   Real party in interest

The real party in interest is Fujitsu Limited, a Japanese corporation with offices at 1-1, Kamikodanaka 4-chome, Nakahara-Ku, Kawasaki-shi, Kanagawa 211-8588, Japan, to which Appellant has assigned all interest in, to and under this application, by virtue of an assignment as recorded at Reel 011024, Frame 0430 of the Assignment records of the U.S. Patent and Trademark Office.

(ii)   Related appeals and interferences

Upon information and belief, there are no other appeals or interferences, which will directly affect or be directly affected by or have a bearing on the Board's decision in this appeal.

(iii)   Status of claims

The application was filed on November 10, 2003 as a continuation application of U.S. Patent Application No. 09/634,389—now U.S. Patent No. 6,687,901, and which was filed on August 9, 2000—and claims foreign priority benefits under 35 U.S.C. §119 based on Japanese Application No. 11-251065 filed on September 6, 1999. The application was filed with claims 12-21—Appellant canceled claims 1-11 and submitted claims 12-21 in a preliminary amendment submitted concurrently with the filing of the application.

In a first Office Action dated October 6, 2006, claim 13 was rejected under 35 U.S.C. § 112, first paragraph, as failing to comply with the enablement requirement; claims 12-13 and 15-16 were rejected under 35 U.S.C. § 101 as being directed to non-statutory subject

84344846_1

matter; claims 12-15 and 18-19 were rejected on the grounds of nonstatutory obviousness-type double patenting as being unpatentable over claims 4-5, 7-8, and 10 of U.S. Patent No. 6,687,901 issued from the parent application; and claims 12-21 were rejected under 35 U.S.C. § 102(e) as being anticipated by U.S. Patent No. 6,023,620 to Hansson.

In a response to the first Office Action, filed on January 4, 2007, Appellant concurrently submitted a terminal disclaimer from U.S. Patent No. 6,687,901 to obviate the double patenting rejection, canceled claims 14 and 20-21, submitted new claim 22, and amended claims 13 and 15-19.

In a non-final Office Action dated March 20, 2007, claims 13, 15-16, 18-19, and 22 were rejected under 35 U.S.C. § 102(e) as being anticipated by Hansson; and claims 16-17 were rejected under 35 U.S.C. § 102(e) as being anticipated by U.S. Patent No. 6,954,754 to Peng.

In a response to the non-final Office Action, filed on June 6, 2007, Appellant amended claims 13 and 17-19, canceled claims 15-16, and submitted new claims 23-32.

In a non-final Office Action dated September 7, 2007, claims 23-25 were rejected under 35 U.S.C. § 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter of the invention; claims 13, 17, 19, and 22 were rejected under 35 U.S.C. § 102(e) as being anticipated by Hansson; claims 18 and 23-32 were rejected under 35 U.S.C. § 102(a) as being anticipated by U.S. Patent No. 5,848,064 to Cowan.

In a response to the non-final Office Action, filed on January 7, 2008, Appellant amended claims 13, 17-19, and 22-32, and submit new claims 33-45.

In a final Office Action dated April 4, 2008, claims 19 and 32 were allowed; and claims 13, 17-18, 22-26, 28-31, and 33-45 were rejected under 35 U.S.C. § 102(a) as being anticipated by Cowan.

84344846_1

In a response to the final Office Action, filed concurrently herewith, Appellant

cancels claims 18, 23-25, 28-31, and 42-44.

The status of the claims for Appeal is:

Claims allowed: 19 and 32

Claims objected to: None

Claims canceled: 1-12, 14-16, 18, 20-21, 23-25, 28-31, and 42-44

Claims withdrawn: None

Claims rejected: 13, 17, 22, 26-27, 33-41, and 45

The rejected claims are set out in the Appendix attached hereto.

The rejected claims are being appealed.

(iv)   **Status of amendments**

Appellant's response filed concurrently with this Brief canceling claims 18, 23-25,

28-31, and 42-44 pursuant to 37 CFR § 41.33(b)(1) is pending at the filing of this Brief.

(v)   **Summary of claimed subject matter**

In conventional radio terminal devices, updating of software requires downloading

update-used software—that is, software data used for updating software in a device—through

a radio communication channel to a programmable ROM in the device. New software is

transmitted from a software-supply device. This conventional technique requires

comparatively large software to be transmitted to and stored at the device, and the update

software often overlaps with current software that is stored in a main memory of the device.

The overlapping software is not only a waste of memory space but also a cause of the

increase in the size of the programmable ROM required for the device. Consequently, it is more difficult to achieve miniaturization and cost reduction of the mobile communication system. Moreover, the ROM must be replaced when software related to control of the radio communication is revised by a revision of communication service.

Furthermore, overwriting a currently operating program with an update-used program has a possibility that the currently operating program might get damaged while the currently operating program downloads the update-used software.

Accordingly, it is a general object of the present invention to provide a method and an apparatus for updating software efficiently and safely with a simple structure and control in a radio terminal device, in which the disadvantages described above are eliminated.

In one embodiment, the present invention provides "[a] software supplying device ['software-supply device 100,' 'software-supply device 50'; Figs. 2-3 and 7-8; page 5, line 14 to page 10, line 21 and page 17, line 25 to page 22, line 14 of the specification] comprising:

a memory ['memory 101,' 'memory 52,' and 'disk 53'; Figs. 2-3; page 5, line 14 to page 10, line 21 of the specification] storing software being downloaded by a radio terminal device ['radio terminal device 200,' 'mobile terminal device 10'; Figs. 2-3; page 5, line 14 to page 10, line 21 of the specification]; and

a communication unit ['communication interface unit 104,' 'communication interface unit 54'; Figs. 2-3; page 5, line 14 to page 10, line 21 of the specification] that is adapted to notify said radio terminal device of a number of divided blocks for transmitting of said stored software ['step J5'; Fig. 7; page 17, line 25 to page 22, line 14 of the specification], to receive from the radio terminal device a request corresponding to each divided block to transmit the respective divided block ['step 45'; Fig. 8; page 17, line 25 to page 22, line 14 of the specification], and to transmit in response to said respective requests said respective divided blocks to the radio terminal device on a block-by-block basis ['steps J7 and S47-S48'; Fig. 8;

page 17, line 25 to page 22, line 14 and page 30, lines 19-34 of the specification]," as recited

in claim 13.

The present invention also provides "[a] radio terminal ['radio terminal device 200,'

'mobile terminal device 10'; Figs. 2-3; page 5, line 14 to page 10, line 21 of the specification]

comprising:

a radio communication unit communicating with a software supplying device

['software-supply device 100,' 'software-supply device 50'; Figs. 2-3 and 7-8; page 5, line 14

to page 10, line 21 and page 17, line 25 to page 22, line 14 of the specification];

a memory storing software presently involved in operations ['main memory 202,'

'main memory 23'; Figs. 2-3; page 5, line 14 to page 10, line 21 of the specification]; and

a controller stopping a download of software from said software supplying device

when the controller detects an operation for responding to an incoming call ['CPU 201,'

'CPU 22'; Figs. 2-3 and 12; page 5, line 14 to page 10, line 21, page 21, lines 30-33, page 29,

line 18 to page 30, line 34 of the specification]," as recited in claim 17.

The present invention provides "[a] software supplying system ['software-supply

device 100,' 'software-supply device 50'; Figs. 2-3 and 7-8; page 5, line 14 to page 10, line

21 and page 17, line 25 to page 22, line 14 of the specification] comprising:

a radio terminal device adapted to transmit and receive communications ['radio

terminal device 200,' 'mobile terminal device 10'; Figs. 2-3; page 5, line 14 to page 10, line

21 of the specification] and including a memory adapted to store a software application; and

a communication unit ['communication interface unit 104,' 'communication interface

unit 54'; Figs. 2-3; page 5, line 14 to page 10, line 21 of the specification] adapted to transmit

to the radio terminal device both a) a number representing a quantity of divided blocks of the

software application ['step J5'; Fig. 7; page 17, line 25 to page 22, line 14 of the

specification], to receive from the radio terminal device a request corresponding to each

84344846_1

divided block to transmit the respective divided block ['step 45'; Fig. 8; page 17, line 25 to page 22, line 14 of the specification], and to transmit in response to said respective requests and b) the respective divided blocks of the software application on a block-by-block basis ['steps J7 and S47-S48'; Fig. 8; page 17, line 25 to page 22, line 14 and page 30, lines 19-34 of the specification]," as recited in claim 22.

The present invention provides "[a] method for updating software in a radio terminal device ['radio terminal device 200,' 'mobile terminal device 10'; Figs. 2-3; page 5, line 14 to page 10, line 21 of the specification], comprising the steps of:

storing, in a memory ['memory 101,' 'memory 52,' and 'disk 53'; Figs. 2-3; page 5, line 14 to page 10, line 21 of the specification], software being downloaded by the radio terminal device;

notifying said radio terminal device of a number of divided blocks for transmitting said stored software ['step J5'; Fig. 7; page 17, line 25 to page 22, line 14 of the specification];

receiving from the radio terminal device a request corresponding to each divided block to transmit the respective divided block ['step 45'; Fig. 8; page 17, line 25 to page 22, line 14 of the specification]; and

transmitting, in response to said respective requests, said respective divided blocks to the radio terminal device on a block-by-block basis ['steps J7 and S47-S48'; Fig. 8; page 17, line 25 to page 22, line 14 and page 30, lines 19-34 of the specification]," as recited in claim 26.

The present invention provides "[a] method for updating software in a radio terminal device ['radio terminal device 200,' 'mobile terminal device 10'; Figs. 2-3; page 5, line 14 to page 10, line 21 of the specification], comprising the steps of:

communicating with a software supplying device ['software-supply device 100,'

'software-supply device 50'; Figs. 2-3 and 7-8; page 5, line 14 to page 10, line 21 and page

17, line 25 to page 22, line 14 of the specification];

storing, in a memory, software presently involved in operations ['main memory 202,'

'main memory 23'; Figs. 2-3; page 5, line 14 to page 10, line 21 of the specification];

detecting whether there is an operation for responding to an incoming call ['CPU

201,' 'CPU 22'; Figs. 2-3 and 12; page 5, line 14 to page 10, line 21, page 21, lines 30-33,

page 29, line 18 to page 30, line 34 of the specification]; and

stopping a download of software from said software supplying device when an

operation for responding to an incoming call is detected ['CPU 201,' 'CPU 22'; Figs. 2-3 and

12; page 5, line 14 to page 10, line 21, page 21, lines 30-33, page 29, line 18 to page 30, line

34 of the specification]," as recited in claim 27.

The present invention provides "[a] software supplying device ['software-supply

device 100,' 'software-supply device 50'; Figs. 2-3 and 7-8; page 5, line 14 to page 10, line

21 and page 17, line 25 to page 22, line 14 of the specification] comprising:

a memory ['memory 101,' 'memory 52,' and 'disk 53'; Figs. 2-3; page 5, line 14 to

page 10, line 21 of the specification] to store software being downloaded by a radio terminal

device ['radio terminal device 200,' 'mobile terminal device 10'; Figs. 2-3; page 5, line 14 to

page 10, line 21 of the specification]; and

a communication unit ['communication interface unit 104,' 'communication interface

unit 54'; Figs. 2-3; page 5, line 14 to page 10, line 21 of the specification] to notify the radio

terminal of a number of N indicating a total number of data blocks ['step J5'; Fig. 7; page

17, line 25 to page 22, line 14 of the specification] and to transmit N data blocks to the radio

terminal in accordance with N requests for transmission of the N data blocks from the radio

terminal ['steps 45, J7, and S47-S48'; Fig. 8; page 17, line 25 to page 22, line 14 and page

84344846_1

30, lines 19-34 of the specification], wherein the N data blocks are components of the software [page 17, line 25 to page 22, line 14 of the specification]," as recited in claim 33.

The present invention provides "[a] software supplying device ['software-supply device 100,' 'software-supply device 50'; Figs. 2-3 and 7-8; page 5, line 14 to page 10, line 21 and page 17, line 25 to page 22, line 14 of the specification] comprising:

a memory ['memory 101,' 'memory 52,' and 'disk 53'; Figs. 2-3; page 5, line 14 to page 10, line 21 of the specification] to store software being downloaded by a radio terminal device ['radio terminal device 200,' 'mobile terminal device 10'; Figs. 2-3; page 5, line 14 to page 10, line 21 of the specification]; and

a communication unit ['communication interface unit 104,' 'communication interface unit 54'; Figs. 2-3; page 5, line 14 to page 10, line 21 of the specification] which performs transmission of the software to the radio terminal by separately transmitting N data blocks which are parts of the software ['steps 45, J7, and S47-S48'; Fig. 8; page 17, line 25 to page 22, line 14 and page 30, lines 19-34 of the specification], wherein the communication unit notifies the number N to the radio terminal before transmitting the N data blocks ['step J5'; Fig. 7; page 17, line 25 to page 22, line 14 of the specification]," as recited in claim 36.

The present invention provides "[a] radio terminal ['radio terminal device 200,' 'mobile terminal device 10'; Figs. 2-3; page 5, line 14 to page 10, line 21 of the specification] comprising:

a radio communication unit ['radio communication unit 205,' elements 11-16; Figs. 2-3; page 5, line 14 to page 10, line 21 of the specification] which communicates with a software supplying device ['software-supply device 100,' 'software-supply device 50'; Figs. 2-3 and 7-8; page 5, line 14 to page 10, line 21 and page 17, line 25 to page 22, line 14 of the specification] to download software stored in the software supplying device; and

a controller which stops a download of the software from the software supplying device

when the controller detects an operation for responding to an incoming call ['CPU 201,' 'CPU 22'; Figs. 2-3 and 12; page 5, line 14 to page 10, line 21, page 21, lines 30-33, page 29, line 18 to page 30, line 34 of the specification]," as recited in claim 39.

The present invention provides "[a] software supplying system comprising:

a radio terminal ['radio terminal device 200,' 'mobile terminal device 10'; Figs. 2-3; page 5, line 14 to page 10, line 21 of the specification] having a radio communication unit ['radio communication unit 205,' elements 11-16; Figs. 2-3; page 5, line 14 to page 10, line 21 of the specification] to communicate with the software supplying device ['software-supply device 100,' 'software-supply device 50'; Figs. 2-3 and 7-8; page 5, line 14 to page 10, line 21 and page 17, line 25 to page 22, line 14 of the specification] via a radio communication line;

a software supplying device having a memory ['memory 101,' 'memory 52,' and 'disk 53'; Figs. 2-3; page 5, line 14 to page 10, line 21 of the specification] to store a software and a communication unit ['communication interface unit 104,' 'communication interface unit 54'; Figs. 2-3; page 5, line 14 to page 10, line 21 of the specification] which notifies the radio terminal of a number which indicates a total number N of data blocks to be downloaded ['step J5'; Fig. 7; page 17, line 25 to page 22, line 14 of the specification] and transmits N data blocks in accordance with requests sent from the radio terminal ['steps 45, J7, and S47-S48'; Fig. 8; page 17, line 25 to page 22, line 14 and page 30, lines 19-34 of the specification], wherein the N data blocks are components of the software [page 17, line 25 to page 22, line 14 of the specification]," as recited in claim 45.

Rejected claims 34-35, 37-38, and 40-41 depend from claims 33, 36, and 39, respectively, and are patentable over the references cited against them for at least the same reasons, as discussed in section (vii).

84344846_1

(vi)   **Grounds of rejection to be reviewed on appeal**

1.   Whether or not claims 13, 17, 22, 26-27, 33-41, and 45 are unpatentable under 35

U.S.C. § 102(a) for being anticipated by U.S. Patent No. 5,848,064 to <u>Cowan</u>.

(vii)   **Argument**

Issue 1:   Whether or not claims 13, 17, 22, 26-27, 33-41, and 45 are unpatentable under 35 U.S.C. § 102(a) for being anticipated by U.S. Patent No. 5,848,064 to Cowan

Appellant respectfully submits that <u>Cowan</u>, as cited and relied upon by the Examiner,

fails to disclose or suggest the features of the claimed invention for which it is relied upon.

Claims 13, 22, 26, 33-38, and 45

Again, the object of the claimed invention is to provide a simplified technique for

updating software efficiently and safely with a simple structure and control in a radio

terminal device. To that end, the claimed invention provides for notification, by a software

supplying device, simply of a <u>number of</u> divided <u>blocks</u> of software downloaded by a radio

terminal device for updating software at the radio terminal device, where the radio terminal

device requests each divided block of software for updating on a block-by-block basis.

Advantageously, the claimed invention provides for simplifying communications in

connection with software updating, and preventing long periods of interruption from normal

operations of the radio terminal device, such as making and receiving calls, while

continuously updating.

In rejecting claim 13—and correspondingly, claims 22, 26, 33-38, and 45—the

Examiner cited description in <u>Cowan</u> of a package definition file that "includes a list of

filenames included in the operating software associated with the package name" as alleged

disclosure of the claimed feature of a software supplying device notifying a radio terminal

device of a number of divided blocks for transmitting stored software being downloaded by

the radio terminal device. Page 3, lines 1-4 of the April 4, 2008 final Office Action (citing

col. 8, lines 57-59 and col. 11, lines 24-35 of Cowan. In particular, the Examiner argued that

"indicating a total number of files through the list," as described in Cowan, disclosed the

claimed divided block number notification feature. *Ibid.*

Appellants respectfully submit that the filename list and the package definition file

described in Cowan clearly fails to disclose the relevant features of the claimed invention.

Indeed, Cowan clearly describes a conventional technique requiring comparatively large

software downloads that is directly addressed by the claimed invention.

As described in Cowan, the package definition file is an omnibus definition file that

defines the total package of software being used by a particular mobile terminal. For

example, Cowan describes

> "when the mobile terminal 36 undergoes its boot-up
> initialization routine the mobile terminal 36 transmits a Bootp
> Request packet 100 to the host computer 30.
> ...
> when the host computer 30 receives the Bootp Request
> packet 100 from the mobile terminal 36, the processor 64
> performs a look-up in the bootptab table stored in the memory
> 66 based on the hardware address contained in the hardware
> address field 103. Specifically, the processor 64 looks up the
> entry in the bootptab table having the same hardware address.
> The processor 64 then takes the corresponding IP address and
> package name from the bootptab table and includes them in the
> IP address field 112 and package name field 113, respectively,
> of the Bootp Response packet 110 which is transmitted to the
> mobile terminal 36." Col. 10, line 35 to col. 11, line 2 of
> Cowan. (Emphasis added)

Based on this Bootp Response packet 110, the mobile terminal saves the package

name, and requests the package definition file corresponding to the package name. Please see

Figs. 7(c) and 7(d) and their corresponding description in Cowan.

In other words, the package definition file is an omnibus information file on the complete software package for use by a particular mobile terminal that is transmitted to the mobile terminal at boot up:

> "Upon receiving the Package Definition Packet 124, the mobile terminal processor 40 initially compares the version identifier included in the package definition field 126 with the version identifier previously stored by the processor 40 the last time the mobile terminal downloaded files from the host computer 30. If the versions are identical, then the mobile terminal 36 concludes that no upgrades or revisions have been performed to the files included in the respective package. If the versions are different, then the mobile terminal 36 stores the information from the Package Definition Packet 124 and then begins the process of sequentially stepping through each file name listed in the package definition file in the package definition field 126 and transmitting a request that the host computer 30 transfer the actual file so that it may be downloaded and stored in the mobile terminal 36." Col. 11, lines 36-51 of <u>Cowan</u>.

And therefore, <u>Cowan</u> only describes transferring a complete package definition file for a mobile terminal when the mobile terminal boots up, the mobile terminal determining whether the package has been updated, and the mobile terminal requesting each file in the package definition file for updating if the package has been updated since the previous boot up check.

<u>Cowan</u>, thus, fails to disclose the claimed feature of a software supplying device notifying or transmitting a "number," quantity," or "value N" of "divided blocks" or "data blocks" "being downloaded" or "to be downloaded" for updating software at a radio terminal device.

Regarding the Examiner's contention that the file list included in the package definition file described in <u>Cowan</u> being adequate disclosure of this feature, Appellant respectfully submits that this file list merely indicates all files included in the complete package of software for a particular mobile terminal, and therefore, does not indicate a

number or quantity of divided blocks to be downloaded, on a block-by-block basis, for

updating software.

The file list described in <u>Cowan</u> merely includes a complete list of file names:

> "As will be appreciated, such file names represent the
> files that are utilized by the mobile terminal 36 to carry out its
> various operations. The actual contents of such files are
> conventional and are not germane to the invention. If, however,
> any of these files are deleted, added, or modified as described
> above, the version indicator is updated in the package
> definition file by the system administrator." Col. 8, lines 59-65
> of <u>Cowan</u>.

Thus, <u>Cowan</u> clearly only describes treating the files on the file list as conventional

discrete files that may be separately "deleted, added, or modified," and fails to disclose any

division of any of the files or the complete package into a number of divided blocks or data

blocks for transmission to a radio terminal—and correspondingly, fails to disclose any

notification to a radio terminal device a number or quantity of such blocks.

Indeed, <u>Cowan</u> describes

> "In the event it is determined that a new version exists,
> every file identified in the package definition file for the
> corresponding package name is downloaded regardless of
> whether only one file or more than one file has been changed,
> added and/or deleted in the package." Col. 12, lines 24-28 of
> <u>Cowan</u>.

And as such, <u>Cowan</u> clearly only describes transferring the complete list of discrete

files for a particular mobile terminal, and allowing the mobile terminal to determine whether

to update the entire package based on this complete information.

Thus, <u>Cowan</u>, as cited and relied upon by the Examiner, fails to disclose,

> "[a] software supplying device comprising:
> a memory storing software <u>being downloaded by a</u>
> <u>radio terminal device</u>; and
> a communication unit that is adapted to <u>notify said</u>
> <u>radio terminal device of *a number of divided blocks*</u> for
> transmitting of said stored software, to receive from the radio
> terminal device a request corresponding to each divided block

84344846_1

to transmit the respective divided block, and to transmit in
response to said respective requests said respective divided
blocks to the radio terminal device on a block-by-block basis,"
as recited in claim 13. (Emphasis added)

Accordingly, Appellant respectfully submits that claim 13 is patentable over <u>Cowan</u>

for at least the above-stated reasons. Claims 22 and 26 incorporate features that correspond

to those of claim 13 cited above, and are, therefore, patentable over <u>Cowan</u> for at least the

same reasons.

Correspondingly, <u>Cowan</u>, as cited and relied upon by the Examiner, fails to disclose,

"[a] software supplying device comprising:
a memory to store software being downloaded by a
radio terminal device; and
a communication unit to <u>notify the radio terminal of *a*</u>
<u>*number of N* indicating a total number of data blocks</u> and to
transmit N data blocks to the radio terminal in accordance with
N requests for transmission of the N data blocks from the radio
terminal, wherein the N data blocks are components of the
software," as recited in claim 33. (Emphasis added)

Accordingly, Appellant respectfully submits that claim 33, together with claims 34-35

dependent therefrom, is patentable over <u>Cowan</u> for at least the above-stated reasons. Claims

36 and 45 incorporate features that correspond to those of claim 33 cited above, and are,

therefore, together with claims 37-38 dependent from claim 36, patentable over <u>Cowan</u> for at

least the same reasons.

<u>Claims 17, 27, and 39-41</u>

With respect to claim 17, the Examiner cited col. 14, lines 1-9 of <u>Cowan</u> as allegedly

disclosing the claimed feature of a radio terminal device stopping a download of software

from a software supplying device <u>when detecting an operation for responding to an incoming</u>

<u>call</u>. Page 3, lines 13-17 of the April 8, 2008 final Office Action. The Examiner simply

stated "terminates if hung up" as allegedly disclosing the claimed feature.

The cited portion of <u>Cowan</u> only includes, however:

> "Regarding steps 154, 158 and 168 discussed above, it will be appreciated that the processor 40 preferably is programmed to <u>retransmit a respective packet</u> only a predetermined number of times (e.g., five), and to <u>terminate</u> the routine <u>in the event a response still is not received</u>. This <u>avoids the mobile terminal becoming hung up due to a system failure.</u> In such case, the processor 40 allows the mobile terminal 36 to continue to operate with the existing version of software stored therein." Col. 14, lines 1-9 of <u>Cowan</u>. (Emphasis added)

In other words, such portion of <u>Cowan</u> only includes description of retransmitting a particular packet a limited number of times so that the process can terminate if a persistent failure remains unresolved. Viewed in the context of the overall system described therein, <u>Cowan</u> only describes limiting a number of retries for a mobile terminal boot up system update so that the mobile terminal does not become "hung up" in such an update—and thus, boot up—process possibly due to a communication or host failure during the update. <u>Cowan</u>, therefore, only describes a technique—limiting the number of retries—that allows the update or boot up process to complete without undue delay caused by update failure. And <u>Cowan</u>, as cited and relied upon by the Examiner, therefore, clearly fails to disclose any detection for any operation for responding to an incoming call as a condition for stopping a download.

Thus, <u>Cowan</u>, as cited and relied upon by the Examiner, clearly fails to disclose,

> "[a] radio terminal comprising:
> a radio communication unit communicating with a
> software supplying device;
> a memory storing software presently involved in
> operations; and
> a controller <u>stopping a download</u> of software from said
> software supplying device <u>when the controller detects an</u>
> <u>operation for responding to an incoming call</u>," as recited in
> claim 17. (Emphasis added)

Accordingly, Appellant respectfully submits that claim 17 is patentable over <u>Cowan</u> for at least the above-stated reasons. Claims 27 and 39 incorporate features that correspond

to those of claim 17 cited above, and are, therefore, together with claims 40-41 dependent

from claim 39, patentable over <u>Cowan</u> for at least the same reasons.

CONCLUSION

In view of the foregoing, Appellant respectfully submits that the Examiner erred in

rejecting claims 13, 17, 22, 26-27, 33-41, and 45, and earnestly requests that this Honorable

Board reverse the Examiner's rejection.

Respectfully submitted,

_____/Dexter T. Chang/_____
Dexter T. Chang
Reg. No. 44,071

CUSTOMER NO.: 026304
Telephone No.: (212) 940-6384
Fax No.: (212) 940-8986/87
Docket No.: FUJI 17.634A (100794-00500)
DTC:tb

84344846_1

(viii) **Claims Appendix**

13.   A software supplying device comprising:

a memory storing software being downloaded by a radio terminal device; and

a communication unit that is adapted to notify said radio terminal device of a number of divided blocks for transmitting of said stored software, to receive from the radio terminal device a request corresponding to each divided block to transmit the respective divided block, and to transmit in response to said respective requests said respective divided blocks to the radio terminal device on a block-by-block basis.

17.   A radio terminal comprising:

a radio communication unit communicating with a software supplying device;

a memory storing software presently involved in operations; and

a controller stopping a download of software from said software supplying device when the controller detects an operation for responding to an incoming call.

22.   A software supplying system comprising:

a radio terminal device adapted to transmit and receive communications and including a memory adapted to store a software application; and

a communication unit adapted to transmit to the radio terminal device both a) a number representing a quantity of divided blocks of the software application, to receive from the radio terminal device a request corresponding to each divided block to transmit the respective divided block, and to transmit in response to said respective requests and b) the respective divided blocks of the software application on a block-by-block basis.

26.	A method for updating software in a radio terminal device, comprising the steps of:

storing, in a memory, software being downloaded by the radio terminal device;

notifying said radio terminal device of a number of divided blocks for transmitting said stored software;

receiving from the radio terminal device a request corresponding to each divided block to transmit the respective divided block; and

transmitting, in response to said respective requests, said respective divided blocks to the radio terminal device on a block-by-block basis.

27.	A method for updating software in a radio terminal device, comprising the steps of:

communicating with a software supplying device;

storing, in a memory, software presently involved in operations;

detecting whether there is an operation for responding to an incoming call; and

stopping a download of software from said software supplying device when an operation for responding to an incoming call is detected.

33.	A software supplying device comprising:

a memory to store software being downloaded by a radio terminal device; and

a communication unit to notify the radio terminal of a number of N indicating a total number of data blocks and to transmit N data blocks to the radio terminal in accordance with N requests for transmission of the N data blocks from the radio terminal, wherein the N data blocks are components of the software.

34.   The software supplying device according to claim 33, wherein the software is

used for updating a present software stored in the radio terminal.

35.   The software supplying device according to claim 33, wherein the software is

used for updating a present control software in the radio terminal.

36.   A software supplying device comprising:

a memory to store software being downloaded by a radio terminal device; and

a communication unit which performs transmission of the software to the radio terminal

by separately transmitting N data blocks which are parts of the software, wherein the

communication unit notifies the number N to the radio terminal before transmitting the N data

blocks.

37.   The software supplying device according to claim 36, wherein the software is

used for updating a present software stored in the radio terminal.

38.   The software supplying device according to claim 36, wherein the software is

used for updating a present control software in the radio terminal.

39.   A radio terminal comprising:

a radio communication unit which communicates with a software supplying device to

download software stored in the software supplying device; and

a controller which stops a download of the software from the software supplying device

when the controller detects an operation for responding to an incoming call.

84344846_1

40. The radio terminal according to claim 39, wherein the software is used for updating a present software stored in the radio terminal.

41. The radio terminal according to claims 39, wherein the software is used for updating a present control software in the radio terminal.

45. A software supplying system comprising:

a radio terminal having a radio communication unit to communicate with the software supplying device via a radio communication line;

a software supplying device having a memory to store a software and a communication unit which notifies the radio terminal of a number which indicates a total number N of data blocks to be downloaded and transmits N data blocks in accordance with requests sent from the radio terminal, wherein the N data blocks are components of the software.

(ix)   **Evidence Appendix**

No evidence was submitted to or entered by the Examiner during prosecution of this application.

(x)   **Related Proceedings Appendix**

Upon information and belief, there are no other appeals or interferences, which will directly affect or be directly affected by or have a bearing on the Board's decision in this appeal.